A

UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/768,098	01/23/2001	David P. Golds	2630	9628

7590	07/18/2005	

LAW OFFICES OF ALBERT S MICHALIK PLLC
704-228TH AVENUE NE
SUITE 193
SAMMAMISH, WA 98074

EXAMINER
BULLOCK JR, LEWIS ALEXANDER

ART UNIT	PAPER NUMBER
2195	

DATE MAILED: 07/18/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _19 April 2005_.

2a)☒ This action is **FINAL**.   2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
   closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-25_ is/are pending in the application.

   4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-25_ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on _23 January 2001_ is/are: a)☒ accepted or b)☐ objected to by the Examiner.

   Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

   Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

   a)☐ All b)☐ Some * c)☐ None of:

   1.☐ Certified copies of the priority documents have been received.

   2.☐ Certified copies of the priority documents have been received in Application No. _____.

   3.☐ Copies of the certified copies of the priority documents have been received in this National Stage
       application from the International Bureau (PCT Rule 17.2(a)).

   * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
   Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
   Paper No(s)/Mail Date. _____.

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 101

1.   35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of
> matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the
> conditions and requirements of this title.

Claim 24 is rejected under 35 U.S.C. 101 because the claimed invention is

directed to non-statutory subject matter. The cited claim details a method comprising a

functional step of classifying software modules into groups by assigning software

modules a static value based on its group. The cited classify operation can be a

defined mental operation wherein a user determines that modules of a certain

functionality are a certain value such that the user determined assignment is stored in a

computer medium. As proper under M.P.E.P. 2106, claims must be both statutory and

containing functional language. The description of a computer medium having non-

functional material, i.e. an association of values to modules, is not statutory.

Claims 21-23 and 25 all detail a computer readable medium for maintaining or

assigning static values to filter drivers and executing the filter drivers in an order

determined by each of the assigned values. The computer readable medium is defined

in the specification to be any available media that **can** be accessed by the computer

and comprises computer storage media and communication media (pg. 8, lines 7-13).

Communication media is defined data structures, program modules, or other data in a

modulated data signal such as a carrier wave or other transport mechanism and

includes any information delivery media (pg. 9, lines 1-12). Therefore, since computer

readable medium includes intangible entities, i.e. wireless media, signals, etc., the

claims are not statutory as detailed in M.P.E.P. 2106 as being directed to non-tangible

subject matter.

Claim Rejections - 35 USC § 102

2.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that

form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

3.   Claims 11, 13, 16-18 and 20 are rejected under 35 U.S.C. 102(e) as being

anticipated by THOMAS (U.S. Patent 6,148,336).

As to claim 11, THOMAS teaches in a computer system (computer architecture)

(col. 7, lines 3-10), a mechanism comprising: a plurality of software modules having a

static assigned value indicative of a relative order (via using keys / parameters / criteria /

class / function / attached filter / predetermined ordering hints to classifying the modules

based on the type of function performed or filter attached); and an ordering mechanism

configured to evaluate each static assigned value and to arrange the software modules

for executing in a relative order determined by the assigned value, the order being

deterministic and static (via ordering the plug-ins and executing them based on a

defined order of function / filter / criteria or ordering hints) (col. 5, lines 7-34; col. 5, lines

45-53; col. 6, lines 59-67; col. 7, lines 45-53; col. 8, line 19-43; col. 10, line 42 – col. 11,

line 58).

As to claim 13, THOMAS teaches the software modules comprise filter drivers

(filters) (col. 5, lines 7-34; col. 5, lines 45-53; col. 6, lines 59-67; col. 7, lines 45-53; col.

8, line 19-43; col. 10, line 42 – col. 11, line 58).

As to claim 16, THOMAS teaches the software modules comprise filter drivers

(filters), and further comprising a filter manager, the filter manager (filter manager)

including the ordering mechanism and further configured to call the filter drivers in the

relative order determined by the assigned values (col. 5, lines 7-34; col. 5, lines 45-53;

col. 6, lines 59-67; col. 7, lines 45-53; col. 8, line 19-43; col. 9, lines 6-13; col. 10, line 42

– col. 11, line 58)..

As to claim 17, THOMAS teaches the filter manager (filter manager) calls the

filter driver (filter) to handle a file system request (via executing the packet) (col. 5, lines

7-34; col. 5, lines 45-53; col. 6, lines 59-67; col. 7, lines 45-53; col. 8, line 19-43; col. 10,

line 42 – col. 11, line 58).

As to claim 18, THOMAS teaches the filter manager (filter manager) is configured

to evaluate criteria associated with the file system request (packet) prior to calling the

filter drivers for execution in the relative order (col. 5, lines 7-34; col. 5, lines 45-53; col.

6, lines 59-67; col. 7, lines 45-53; col. 8, line 19-43; col. 10, line 42 – col. 11, line 58).

As to claim 20, THOMAS teaches each assigned value is unique to a particular

software module (via the information is used to determine whether the plug-in is to be

assigned) (col. 11, lines 10-58).

Claim Rejections - 35 USC § 103

4.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5.   Claims 1-10, 19 and 21-25 are rejected under 35 U.S.C. 103(a) as being

unpatentable over THOMAS (U.S. Patent 6,148,336) in view of "Router Plugins,

Asoftware Architecture for Next Generation Routers" by DECASPER et al.

As to claim 1, THOMAS teaches in a computer system (computer architecture)

(col. 7, lines 3-10), a method comprising: maintaining static assigned values (keys /

parameters / criteria / class / function / attached filter / predetermined ordering hints) in

association with software modules (plug-ins / filters), each software module having a

static assigned value, the assigned values having a relative order (via classifying the

modules based on the type of function performed or filter attached); and executing the

software modules in an order determined by each of the assigned values, the order

being deterministic and static (via ordering the plug-ins and executing them based on a

defined order) (col. 5, lines 7-34; col. 5, lines 45-53; col. 6, lines 59-67; col. 7, lines 45-

53; col. 8, line 19-43; col. 10, line 42 – col. 11, line 58). However, THOMAS does not

teach there being an unassigned value between any two assigned values.

DECASPER teaches the dynamic loading and managing of software modules

(plug-ins and their associated filters) (pg. 231) wherein the software modules have a

static assigned value (source address / destination address to handle a particular

protocol) (see pg. 235, table 1). It would be obvious based on the teachings of disclosed

on page 235, that there exists unassigned values between any two assigned values

(wherein filter handles TCP communications from 129.* to 192.94.233.10 and filter 3

handles TCP communications from 128.252.153.1 – 128.252.153.7, therefore,

128.252.153.8 – 129 are unassigned). Therefore, it would be obvious to combine the

teachings of THOMAS with the teachings of DECASPER in order to facilitate the ability

to bind different plug-ins to individual flows thereby allowing for distinct plug-in

implementations to seamlessly coexist in the same runtime environment (abstract).

As to claim 21, reference is made to a computer medium that corresponds to the

method of claim 1 and is therefore met by the rejection of claim 1 above.

As to claims 24 and 25, THOMAS teaches a method comprising: classifying

software modules into groups based on types thereto (via ordering of plug-ins by class,

function, attached filter, and predetermined ordering hints); assigning and maintaining

static assigned values (keys / parameters / criteria / class / function / attached filter / predetermined ordering hints) in association with software modules (plug-ins / filters), each software module having a static assigned value, the assigned values having a relative order (via classifying the modules based on the type of function performed or filter attached); and executing the software modules in an order determined by each of the assigned values, the order being deterministic and static (via ordering the plug-ins and executing them based on a defined order) (col. 5, lines 7-34; col. 5, lines 45-53; col. 6, lines 59-67; col. 7, lines 45-53; col. 8, line 19-43; col. 10, line 42 – col. 11, line 58). However, THOMAS does not teach there being an unassigned value between any two assigned values.

DECASPER teaches the dynamic loading and managing of software modules (plug-ins and their associated filters) (pg. 231) wherein the software modules have a static assigned value (source address / destination address to handle a particular protocol) (see pg. 235, table 1). It would be obvious based on the teachings of disclosed on page 235, that there exists unassigned values between any two assigned values (wherein filter handles TCP communications from 129.* to 192.94.233.10 and filter 3 handles TCP communications from 128.252.153.1 – 128.252.153.7, therefore, 128.252.153.8 – 129 are unassigned). Therefore, it would be obvious to combine the teachings of THOMAS with the teachings of DECASPER in order to facilitate the ability to bind different plug-ins to individual flows thereby allowing for distinct plug-in implementations to seamlessly coexist in the same runtime environment (abstract).

As to claim 2, THOMAS teaches executing the software modules (col. 10, line 64 – col. 11, line 58; abstract). It would be obvious to one skilled in the art at the time of the invention that in order to execute the software modules, they would have to be called.

As to claims 3 and 4, THOMAS teaches the software modules comprise filter drivers (filters), and wherein calling the software modules includes passing packets thereto (col. 5, lines 7-34; col. 5, lines 45-53; col. 6, lines 59-67; col. 7, lines 45-53; col. 8, line 19-43; col. 10, line 42 – col. 11, line 58). However, THOMAS does not teach the packets are input-output packets or file system packets. Official Notice is taken in that it is well known in the art that input-output request or file system request are embedded in packets and therefore it would be obvious to one skilled in the art at the time of the invention that the THOMAS would handle these packets as well

As to claim 5, THOMAS teaches the software modules are loaded and attached in the system (col. 8, lines 12-26; fig. 6). However, THOMAS does not explicitly state that the modules are in a stack format. Official Notice is taken in that stack formats are well known in the art and therefore would be obvious to one skilled in the art at the time of the invention to load the software modules in a stack format.

As to claim 6, THOMAS teaches executing the software modules in an order

determined by each of the assigned values includes maintaining an order (col. 5, lines

7-34; col. 5, lines 45-53; col. 6, lines 59-67; col. 7, lines 45-53; col. 8, line 19-43; col. 10,

line 42 – col. 11, line 58).

As to claim 7, THOMAS teaches evaluating criteria associated with the software

modules, and wherein executing the software modules comprises selecting only

software module that meet the criteria for execution (enable or disabling plug-ins /

selecting plug-ins that apply) (col. 5, lines 7-34; col. 5, lines 45-53; col. 6, lines 59-67;

col. 7, lines 45-53; col. 8, line 19-43; col. 10, line 42 – col. 11, line 58).

As to claim 8, THOMAS teaches the software modules comprise filter drivers

(filters), and wherein evaluating criteria associated with the software modules comprises

evaluating a packet (via selecting the plug-in or filter that corresponds to the packet)

(col. 5, lines 7-34; col. 5, lines 45-53; col. 6, lines 59-67; col. 7, lines 45-53; col. 8, line

19-43; col. 10, line 42 – col. 11, line 58). However, THOMAS does not teach the

packets are input-output packets or file system packets. Official Notice is taken in that it

is well known in the art that input-output request or file system request are embedded in

packets and therefore it would be obvious to one skilled in the art at the time of the

invention that the THOMAS would handle these packets as well

As to claim 9, THOMAS teaches assigning an assigned value to a software module (col. 5, lines 7-34; col. 5, lines 45-53; col. 6, lines 59-67; col. 7, lines 45-53; col. 8, line 19-43; col. 10, line 42 – col. 11, line 58). DECASPER also teaches assigning an assigned value to a software module (pg. 234, "Each plugin in our framework is identified by a 32 bit plugin code. The upper 16 bits of the code identify the plugin type...").

As to claim 10, THOMAS teaches classifying a software module based on a type thereof, and wherein the assigned value corresponds to the type (col. 10, lines 41 – col. 11, line 58).

As to claim 19, THOMAS substantially discloses the invention above. However, THOMAS does not teach there being an unassigned value between any two assigned values. DECASPER teaches the dynamic loading and managing of software modules (plug-ins and their associated filters) (pg. 231) wherein the software modules have a static assigned value (source address / destination address to handle a particular protocol) (see pg. 235, table 1). It would be obvious based on the teachings of disclosed on page 235, that there exists unassigned values between any two assigned values (wherein filter handles TCP communications from 129.* to 192.94.233.10 and filter 3 handles TCP communications from 128.252.153.1 – 128.252.153.7, therefore, 128.252.153.8 – 129 are unassigned). Therefore, it would be obvious to combine the teachings of THOMAS with the teachings of DECASPER in order to facilitate the ability

to bind different plug-ins to individual flows thereby allowing for distinct plug-in

implementations to seamlessly coexist in the same runtime environment (abstract).

As to claim 22, THOMAS teaches executing the filter drivers in an order

determined by each of the assigned values (via priority indications) and passing the

packets to the filter drivers (col. 5, lines 7-34; col. 5, lines 45-53; col. 6, lines 59-67; col.

7, lines 45-53; col. 8, line 19-43; col. 10, line 42 – col. 11, line 58). However, THOMAS

does not explicitly state that the modules are in a stack format or that the packets

passed to the filter drivers are file system requests. Official Notice is taken in that stack

formats are well known in the art and therefore would be obvious to one skilled in the art

at the time of the invention to load the software modules in a stack format. Official

Notice is also taken in that it is well known in the art that input-output request or file

system request are embedded in packets and therefore it would be obvious to one

skilled in the art at the time of the invention that the THOMAS would handle these

packets as well

As to claim 23, THOMAS teaches executing the filter drivers (filters) in an order

determined by each of the assigned values (priority indications) includes calling the filter

drivers in the order determined by each of the assigned values to pass file system

request thereto (col. 5, lines 7-34; col. 5, lines 45-53; col. 6, lines 59-67; col. 7, lines 45-

53; col. 8, line 19-43; col. 10, line 42 – col. 11, line 58).

6.   Claims 12, 14, and 15 are rejected under 35 U.S.C. 103(a) as being

unpatentable over THOMAS (U.S. Patent 6,148,336).

As to claim 12, THOMAS teaches the software modules are loaded and attached

in the system (col. 8, lines 12-26; fig. 6). However, THOMAS does not explicitly state

that the modules are in a stack format. Official Notice is taken in that stack formats are

well known in the art and therefore would be obvious to one skilled in the art at the time

of the invention to load the software modules in a stack format.

As to claims 14 and 15, THOMAS teaches the software modules are configured

to handle packets (network packets) (col. 8, lines 5-10). However, THOMAS does not

teach the packets are input-output packets or file system packets. Official Notice is

taken in that it is well known in the art that input-output request or file system request

are embedded in packets and therefore it would be obvious to one skilled in the art at

the time of the invention that the THOMAS would handle these packets as well.

Response to Arguments

7.   Applicant's arguments with respect to claims 1-25 have been considered but are

moot in view of the new ground(s) of rejection.

Conclusion

8.   Applicant's amendment necessitated the new ground(s) of rejection presented in

this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP

§ 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37
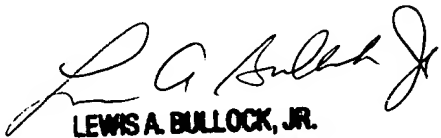CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE
MONTHS from the mailing date of this action. In the event a first reply is filed within
TWO MONTHS of the mailing date of this final action and the advisory action is not
mailed until after the end of the THREE-MONTH shortened statutory period, then the
shortened statutory period will expire on the date the advisory action is mailed, and any
extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of
the advisory action. In no event, however, will the statutory period for reply expire later
than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the
examiner should be directed to Lewis A. Bullock, Jr. whose telephone number is (571)
272-3759. The examiner can normally be reached on Monday-Friday, 8:30 - 5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's
supervisor, Meng An can be reached on (571) 272-3756. The fax phone number for the
organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

LEWIS A. BULLOCK, JR.
PRIMARY EXAMINER

July 13, 2005